EURASIP Journal on Applied Signal Processing 2004:15, 2385–2395

Particle Filtering Applied to Musical Tempo Tracking

Stephen W. Hainsworth
Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK
Email: swh21@cantab.net

Malcolm D. Macleod
QinetiQ, Malvern, WR14 3PS, UK
Email: m.macleod@signal.qinetiq.com

Received 30 May 2003; Revised 1 May 2004

This paper explores the use of particle filters for beat tracking in musical audio examples. The aim is to estimate the time-varying tempo process and to find the time locations of beats, as defined by human perception. Two alternative algorithms are presented, one which performs Rao-Blackwellisation to produce an almost deterministic formulation while the second is a formulation which models tempo as a Brownian motion process. The algorithms have been tested on a large and varied database of examples and results are comparable with the current state of the art. The deterministic algorithm gives the better performance of the two algorithms.

Keywords and phrases: beat, tracking, particle filters, music.

1. INTRODUCTION

Musical audio analysis has been a growing area for research over the last decade. One of the goals in the area is fully automated transcription of real polyphonic audio signals, though this problem is currently only partially solved. More realistic sub-tasks in the overall problem exist and can be explored with greater success; beat tracking is one of these and has many applications in its own right (automatic accompaniment of solo performances [1], auto-DJs, expressive rhythmic transformations [2], uses in database retrieval [3], metadata generation [4], etc.).

This paper describes an investigation into beat tracking utilising particle filtering algorithms as a framework for sequential stochastic estimation where the state-space under consideration is a complex one and does not permit a closed form solution.

Historically, a number of methods have been used to attempt solution of the problem, though they can be broadly categorised into a number of distinct methodologies.[1] The oldest approach is to use oscillating filterbanks and to look for the maximum output; Scheirer [7] typifies this approach though Large [8] is another example. Autocorrelative methods have also been tried and Tzanetakis [3] or Foote [9] are examples, though these tend to only find the average tempo and not the phase (as defined in Section 2) of the beat. Multiple hypothesis approaches (e.g., Goto [10] or Dixon [11]) are very similar to more rigorously probabilistic approaches (Laroche [12] or Raphael [13], for instance) in that they all evaluate they likelihood of a hypothesis set; only the framework varies from case to case. Klapuri [14] also presents a method for beat tracking which takes the approach typified by Scheirer [7] and applies a probabilistic tempo smoothness model to the raw output. This is tested on an extensive database and the results are the current state of the art.

More recently, particle filters have been applied to the problem; Morris and Sethares [15] briefly present an algorithm which extracts features from the signal and then uses these feature vectors to perform sequential estimation, though their implementation is not described. Cemgil [16] also uses a particle filtering method in his comprehensive paper applying Monte Carlo methods to the beat tracking of expressively performed MIDI signals.[2] This model will be discussed further later, as it shares some aspects with one of the models described in this paper.

The remainder of the paper is organised as follows: Section 2 introduces tempo tracking; Section 3 covers basic

[1] A comprehensive literature review can be found in Seppänen [5] or Hainsworth [6].

[2] MIDI stands for "musical instrument digital interface" and is a language for expressing musical events in binary. In the context described here, the note start times are extracted from the MIDI signal.

particle filtering theory. Sections 4, 5 and 6 discuss onset detection and the two beat tracking models proposed. Results and discussion are presented in Sections 7 and 8, and conclusions in Section 9.

2. TEMPO TRACKING AND BEAT PERCEPTION

So what is beat tracking?[3] The least jargon-ridden description is that it is the pulse defined by a human listener tapping in time to music. However, the terms *tempo*, *beat* and *rhythm* need to be defined. The highest level descriptor is the rhythm; this is the full description of every timing relationship inside a piece of music. However, Bilmes [17] breaks this down into four subdivisions: the hierarchical *metrical structure* which describes the idealised timing relationships between musical events (as they might exist in a musical score for instance), *tempo variations* which link these together in a possibly time varying flow, *timing deviations* which are individual timing discrepancies ("swing" is an example of this) and finally *arrhythmic sections*. If one ignores the last of these as fundamentally impossible to analyse meaningfully, the task is to estimate the tempo curve (tempo tracking) and idealised event times quantised to a grid of "score locations," given an input set of musical changepoint times.

To represent the tempo curve, a frequency and phase is required such that the phase is zero at beat locations. The metrical structure can then be broken down into a set of levels described by Klapuri [14]: the *beat* or *tactus* is the preferred human tapping tempo; the *tatum* is the shortest commonly occurring interval; and the *bar* or *measure* is related to harmonic change and often correlates to the bar line in common score notation of music. It should be noted that the beat often corresponds to the 1/4 note or crotchet in common notation, but this is not always the case: in fast jazz music, the beat is often felt at half this rate; in hymn music, traditional notation often gives the beat two crotchets (i.e., 1/2 note). The moral is that one must be careful about relating perception to musical notation! Figure 1 gives a diagrammatic representation of the beat relationships for a simple example. The beat is subdivided by two to get the tatum and grouped in fours to find the bar. The lowest level shows timing deviations around the fixed metrical grid.

Perception of rhythm by humans has long been an active area of research and there is a large body of literature on the subject. Drake et al. [18] found that humans with no musical training were able to tap along to a musical audio sample "in time with the music," though trained musicians were able to do this more accurately. Many other studies have been undertaken into perception of simple rhythmic patterns (e.g., Povel and Essens [19]) and various models of beat perception have been proposed (e.g., [20, 21, 22]) from which ideas can be gleaned. However, the models presented in the rest of this paper are not intended as perceptual models or even as perceptually motivated models; they are engineering equivalents of the human perception. Having said that, it is hoped that a successful computer algorithm could help shed light onto potential and as yet unexplained human cognitive processes.

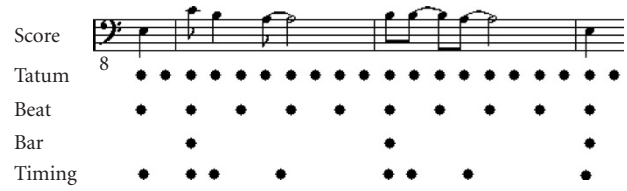

Figure 1: Diagram of relationships between metrical levels.

2.1. Problem statement

To summarise, the aim of this investigation is to extract the beat from music as defined by the preferred human tapping tempo; to make the computer tap its hypothetical foot along in time to the music. This requires a tempo process to be explicitly estimated in both frequency and phase, a beat lying where phase is zero. In the process of this, detected "notes" in the audio are assigned "score locations" which is equivalent to quantising them to an underlying, idealised metrical grid. We are not interested in real time implementation nor in *causal* beat tracking where only data up to the currently considered time is used for estimation.

3. PARTICLE FILTERING

Particle filters are a sequential Monte Carlo estimation method which are powerful, versatile and increasingly used in tracking problems. Consider the state space system defined by

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}, \xi_k), \tag{1}$$

where $f_k : \Re^{n_x} \times \Re^{n_\xi} \to \Re^{n_x}, k \in \mathbb{N}$, is a possibly nonlinear function of the state $\mathbf{x}_{k-1}$, dimension n_x and ξ_k which is an i.i.d. noise process of dimension n_ξ. The objective is to estimate $\mathbf{x}_k$ given observations,

$$\mathbf{y}_k = h_k(\mathbf{x}_k, \nu_k), \tag{2}$$

where $h_k : \Re^{n_x} \times \Re^{n_\nu} \to \Re^{n_y}$ is a separate possibly nonlinear transform and ν_k is a separate i.i.d. noise process of dimension n_ν describing the observation error.

The posterior of interest is given by $p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$ which is represented in particle filters by a set of point estimates or particles $\{\mathbf{x}_{0:k}^{(i)}, w_k^{(i)}\}_{i=1}^N$, where $\{\mathbf{x}_{0:k}^{(i)},\ i = 1,\ldots,N\}$ is a set of support points with associated weights given by $\{w_k^{(i)},\ i = 1,\ldots,N\}$. The weights are normalised such that $\sum_{i=1}^N w_k^{(i)} = 1$. The posterior is then approximated by

$$p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^{(i)}). \tag{3}$$

[3] A fuller discussion on this topic can be found in [6].

As $N \to \infty$, this assumption asymptotically tends to the true posterior. The weights are then selected according to *importance sampling*, $\mathbf{x}_{0:k}^{(i)} \sim \pi(\mathbf{x}_{0:k}^{(i)}|\mathbf{y}_{1:k})$, where $\pi(\cdot)$ is the so-called importance density. The weights are then given by

$$w_k^{(i)} \propto \frac{p(\mathbf{x}_{0:k}^{(i)}|\mathbf{y}_{1:k})}{\pi(\mathbf{x}_{0:k}^{(i)}|\mathbf{y}_{1:k})}. \tag{4}$$

If we restrict ourselves to importance functions of the form,

$$\pi(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) = \pi(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})\pi(\mathbf{x}_{0:k-1}|\mathbf{y}_{1:k-1}), \tag{5}$$

implying a Markov dependency of order 1, the posterior can be factorised to give

$$\begin{aligned} &p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) \\ &= \frac{p(\mathbf{y}_k|\mathbf{x}_{0:k}, \mathbf{y}_{1:k-1})\, p(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1})} \times p(\mathbf{x}_{0:k-1}|\mathbf{y}_{1:k-1}) \\ &\propto p(\mathbf{y}_k|\mathbf{x}_{0:k}, \mathbf{y}_{1:k-1})\, p(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{1:k-1})\, p(\mathbf{x}_{0:k-1}|\mathbf{y}_{1:k-1}), \end{aligned} \tag{6}$$

which allows sequential update. The weights can then be proven to be updated [23] according to

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})\, p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)})}{\pi(\mathbf{x}_k^{(i)}|\mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})} \tag{7}$$

up to a proportionality. Often we are interested in the filtered estimate $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ which can be approximated by

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}). \tag{8}$$

Particle filters often suffer from degeneracy as all but a small number of weights drop to almost zero, a measure of this being approximated by $\widehat{N_{\text{eff}}} = 1/\sum_{i=1}^{N}(w_k^{(i)})^2$ [23]. Good choice of the importance density $\pi(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})$ can delay this and is crucial to general performance. The introduction of a stochastic jitter into the particle set can also help [24]; however the most common solution is to perform resampling [25] whereby particles with small weights are eliminated and a new sample set $\{\mathbf{x}_k^{(i)*}\}_{i=1}^{N}$ is generated by resampling N times from the approximate posterior as given by (8) such that $\Pr(\mathbf{x}_k^{(i)*} = \mathbf{x}_k^{(j)}) = w_k^{(j)}$. The new sample set is then more closely distributed according to the true posterior and the weights should be set to $w_k^{(i)} = 1/N$ to reflect this. Further details on particle filtering can be found in [23, 26].

A special case of model is the jump Markov linear systems (JMLS) [27] where the state space, $\mathbf{x}_{0:k}$, can be broken down into $\{\mathbf{r}_{0:k}, \mathbf{z}_{0:k}\}$. $\mathbf{r}_{0:k}$, the jump Markov process, defines a path through a bounded and discrete set of potential states and conditional upon $\mathbf{r}_{0:k}$, $\mathbf{z}_{0:k}$ is then defined to be linear Gaussian. The chain rule gives the expansion,

$$p(\mathbf{r}_{0:k}, \mathbf{z}_{0:k}|\mathbf{y}_{1:k}) = p(\mathbf{z}_{0:k}|\mathbf{r}_{0:k}, \mathbf{y}_{1:k})\, p(\mathbf{r}_{0:k}|\mathbf{y}_{1:k}), \tag{9}$$

and $p(\mathbf{x}_{0:k}|\mathbf{r}_{0:k}, \mathbf{y}_{1:k})$ is deterministically evaluated via the Kalman filter equations given below in Section 5. After this marginalisation process (called Rao-Blackwellisation [28]), $p(\mathbf{r}_{0:k}|\mathbf{y}_{1:k})$ is then expanded as

$$\begin{aligned} &p(\mathbf{r}_{0:k}|\mathbf{y}_{1:k}) \\ &\quad \propto p(\mathbf{y}_k|\mathbf{r}_{0:k}, \mathbf{y}_{1:k-1})\, p(\mathbf{r}_k|\mathbf{r}_{k-1}) \times p(\mathbf{r}_{0:k-1}|\mathbf{y}_{1:k-1}), \end{aligned} \tag{10}$$

with associated (unnormalised) importance weights given by

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k|\mathbf{r}_{0:k}^{(i)}, \mathbf{y}_{1:k-1})\, p(\mathbf{r}_k^{(i)}|\mathbf{r}_{k-1}^{(i)})}{\pi(\mathbf{r}_k^{(i)}|\mathbf{r}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})}. \tag{11}$$

By splitting the state space up in this way, the dimensionality considered in the particle filter itself is dramatically decreased and the number of particles needed to achieve a given accuracy is also significantly reduced.

4. CHANGE DETECTION

The success of any algorithm is dependent upon the reliability of the data which is provided as an input. Thus, detecting note events in the music for the particle filtering algorithms to track is as important as the actual algorithms themselves. The onset detection falls into two categories; firstly there is detection of transient events which are associated with strong energy changes, epitomised by drum sounds. Secondly, there is detection of harmonic changes without large associated energy changes (e.g., in a string quartet). To implement the first of these, our method approximately follows many algorithms in the literature [7, 11, 12]: frequency bands, f, are separated and an energy evolution envelope $E_f(n)$ formed. A three point linear regression is used to find the gradient of $E_f(n)$ and peaks in this gradient function are detected (equivalent to finding sharp, positive increases in energy which hopefully correspond to the start of notes). Low-energy onsets are ignored and when there are closely spaced pairs of onsets, the lower amplitude one is also discarded. Three frequency bands were used: 20–200 Hz to capture low frequency information; 200 Hz–15 kHz which captures most of the harmonic spectral region; and 15–22 kHz which, contrary to the opinion of Duxbury [29], is generally free from harmonic sounds and therefore clearly shows any transient information.

Harmonic change detection is a harder problem and has received very little attention in the past, though two recent studies have addressed this [29, 30]. To separate harmonics in the frequency domain, long short-time Fourier transform (STFT) windows (4096 samples) with a short hop rate (1/8 frame) were used. As a measure of spectral change from one

frame to the next, a modified Kullback-Liebler distance measure was used:

$$d_n(k) = \log_2\left(\frac{|X[k,n]|}{|X[k,n-1]|}\right),$$
$$D_{\mathrm{MKL}}(n) = \sum_{k\in\mathcal{K},\, d(n)>0} d_n(k), \tag{12}$$

where $X[k,n]$ is the STFT with time index n and frequency bin k. The modified measure is thus tailored to accentuate positive energy change. $\mathcal{K}$ defines the region 40 Hz–5 kHz where the majority of harmonic energy is to be found and to pick peaks, a local average of the function D_{MKL} was formed and then the maximum picked between each of the crossings of the actual function and the average.

A further discussion of the MKL measure can be found in [31] but a comprehensive analysis is beyond the scope of this paper. For beat tracking purposes, it is desirable to have a low false detection rate, though missed detections are not so important. While no actual rates for false alarms have been determined, the average detected inter-onset interval (IOI) was compared with an estimate given by $T/(N_b \times F)$, where T is the length of the example in seconds, N_b is the number of manually labelled beats and F is the number of tatums in a beat. The detected average IOI was always of the order or larger than the estimate, which shows that under-detection is occurring.

In summary, there are four vectors of onset observations, three from energy change detectors and one from a harmonic change detector. The different detectors may all observe an actual note, or any combination of them might not. In fact, clustering of the onset observations from each of the individual detection functions is performed prior to the start of the particle filtering. A group is formed if any events from different streams fall within 50 ms of each other for transient onsets and 80 ms for harmonic onsets (reflecting the lower time resolution inherent in the harmonic detection process). Inspection of the resulting grouped onsets shows that the inter-group separation is usually significantly more than the within-group time differences. A set of amplitudes is then associated with each onset cluster.

5. BEAT MODEL 1

The model used in this section is loosely based on that of Cemgil et al. [16], designed for MIDI signals. Given the series of onset observations generated as above, the problem is to find a tempo profile which links them together and to assign each observation to a quantised score location.

The system can be represented as a JMLS where conditional on the "jump" parameter, the system is linear Gaussian and the traditional Kalman filter can be used to evaluate the sequence likelihood. The system equations are then

$$\mathbf{x}_k = \Phi_k(\gamma_k)\mathbf{x}_{k-1} + \boldsymbol{\xi}_k, \tag{13}$$
$$\mathbf{y}_k = H_k\mathbf{x}_k + \nu_k, \tag{14}$$

where $\mathbf{x}_k$ is the tempo process at iteration k and can be described as $\mathbf{x}_k = [\rho_k, \Delta_k]^T$. ρ_k is then the predicted time of the kth observation and Δ_k the tempo period, that is, $\Delta_k = 60/T_k$, where T_k is the tempo in beats per minute (bpm). This is equivalent to a constant velocity process and the state innovation, ξ_k is modelled as zero mean Gaussian with covariance Q_k.

To solve the quantisation problem, the score location is encoded as the jump parameter, γ_k, in $\Phi_k(\gamma_k)$. This is equivalent to deciding upon the notation that describes the rhythm of the observed notes. $\Phi_k(\gamma_k)$, is then given by

$$\Phi_k(\gamma_k) = \begin{bmatrix} 1 & \gamma_k \\ 0 & 1 \end{bmatrix},$$
$$\gamma_k = c_k - c_{k-1}. \tag{15}$$

This associated evolution covariance matrix is [32]

$$Q_k = q\begin{bmatrix} \frac{\gamma_k^3}{3} & \frac{\gamma_k^2}{2} \\ \frac{\gamma_k^2}{2} & \gamma_k \end{bmatrix}, \tag{16}$$

for a continuous constant velocity process which is observed at discrete time intervals, where q is a scale parameter.

While the state transition matrix is dependent upon γ_k, this is a difference term between two actual locations, c_k and c_{k-1}. It is this process which is important and the prior on c_k becomes a critical issue as it determines the performance characteristics. Cemgil breaks a single beat into subdivisions of two and uses a prior related to the number of significant digits in the binary expansion of the quantised location. Cemgil's application was in MIDI signals where there is 100% reliability in the data and the onset times are accurate. In audio signals, the event detection process introduces errors both in localisation accuracy and in generating entirely spurious events. Also, Cemgil's prior cannot cope with triplet figures or swing. Thus, we break the notated beat down into 24 quantised sub-beat locations, $c_k = \{1/24, 2/24, \ldots, 24/24, 25/24, \ldots\}$ and assign a prior

$$p(c_k) \propto \exp\left(-\log_2\{d(c_k)\}\right), \tag{17}$$

where $d(c_k)$ is the denominator of the fraction of c_k when expressed in its most reduced form; that is, $d(3/24) = 8$, $d(36/24) = 2$, and so forth. This prior is motivated by the simple concern of making metrically stronger sub-beat locations more likely; it is a generic prior designed to work with all styles and situations.

Finally, the observation model must be considered. Bearing in mind the pre-processing step of clustering onset observations from different observation function, the input to the particle filter at each step $\mathbf{y}_k$ will be a variable length vector containing between one and four individual onset observation times. Thus, H_k becomes a function of the length j of the observation vector $\mathbf{y}_k$ but is essentially j rows of the form $[1\ \ 0]$. The observation error ν_k is also of length j and

is modelled as zero-mean Gaussian with diagonal covariance R_k where the elements r_{jj} are related to whichever observation vector is being considered at $\mathbf{y}_k(j)$.

Thus, conditional upon the c_k process which defines the update rate, everything is modelled as linear Gaussian and the traditional Kalman filter [33] can be used. This is given by the recursion

$$\begin{aligned}\hat{\mathbf{x}}_{k|k-1} &= \Phi_k \hat{\mathbf{x}}_{k-1|k-1},\\ P(k|k-1) &= \Phi_k P(k-1|k-1)\Phi_k^T + Q_k,\\ K(k) &= P(k|k-1)H_k^T\left[H_k P(k|k-1)H_k^T + R_k\right]^{-1},\\ \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + K(k)\left[\mathbf{y}_k - H_k\hat{\mathbf{x}}_{k|k-1}\right],\\ P(k|k) &= \left[I - K(k)H_k\right]P(k|k-1).\end{aligned} \tag{18}$$

Each particle must maintain its own covariance estimate $P(k|k)$ as well as its own state. The innovation or residual vector is defined to be the difference between the measured and predicted quantities,

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - H_k\hat{\mathbf{x}}_{k|k-1}, \tag{19}$$

and has covariance given by

$$S_k = H_k P_{k|k-1}H_k^T + R_k. \tag{20}$$

5.1. Amplitude modelling

The algorithm as described so far will assign the beat (i.e., the phase of $c_{1:k}$) to the most frequent subdivision, which may not be the right one. To aid the correct determination of phase, attention is turned to the amplitude of the onsets.

The assumption is made that the onsets at some score locations (e.g., on the beat) will have higher energy than others. Each of the three transient onset streams maintains a separate amplitude process while the harmonic onset stream does not have one associated with it due to amplitude not being relevant for this feature.

The amplitude processes can be represented as separate JMLSs conditional upon c_k. The state equations are given by

$$\begin{aligned}\alpha_p^l &= \Theta_p^l \alpha_{p-1}^l + \epsilon_p,\\ a_p^l &= \alpha_p^l + \varsigma_p,\end{aligned} \tag{21}$$

where a_p^l is the amplitude of the pth onset from the observation stream, l. Thus, the individual process is maintained for each observation function and updated only when a new observation from that stream is encountered. This requires the introduction of conditioning on p rather than k; $1{:}p$ then represents all the indices within the full set $1{:}k$, where an observation from stream l is found. $\Theta_p^l(c_{p-1}, c_p)$ is a function of c_p and c_{p-1}. To build up the matrix, Θ_p^l, a selection of real data was examined and a 24×24 matrix constructed for the expected amplitude ratio between a pair of score locations. This is then indexed by the currently considered score location c_p and also the previously identified one found in stream l, c_{p-1}^l, and the value given is returned to Θ_p^l. For example, it could be that the expected amplitude for a beat is modelled as twice that of a quaver off-beat. If the particle history shows that the previous onset from a given stream was assigned to be on the beat and the currently considered location is a quaver, Θ_p^l would equal 0.5. This relative relationship allows the same model to cope with both quiet and loud sections in a piece. The evolution and observation error terms, ϵ_p and ς_p, are assumed to be zero mean Gaussian with appropriate variances.

From now on, to avoid complicating the notation, the amplitude process will be represented without sums or products over the three l vectors using $a_p = \{a_p^1, a_p^2, a_p^3\}$ and $\alpha_p = \{\alpha_p^1, \alpha_p^2, \alpha_p^3\}$ (noting that some of these might well be given a null value at any given iteration). For each iteration k, between zero and all three of the amplitude processes will be updated.

5.2. Methodology

Given the above system, a particle filtering algorithm can be used to estimate the posterior at any given iteration. The posterior which we wish to estimate is given by $p(c_{1:k}, \mathbf{x}_{1:k}, \alpha_{1:p}|\mathbf{y}_{1:k}, a_{1:p})$ but Rao-Blackwellisation breaks down the posterior into separate terms

$$\begin{aligned}&p(c_{1:k}, \mathbf{x}_{1:k}, \alpha_{1:p}|\mathbf{y}_{1:k}, a_{1:p})\\ &\quad= p(\mathbf{x}_{1:k}|c_{1:k}, \mathbf{y}_{1:k})\\ &\qquad\times p(\alpha_{1:p}|c_{1:k}, a_{1:p})\,p(c_{1:k}|\mathbf{y}_{1:k}, a_{1:p}),\end{aligned} \tag{22}$$

where $p(\mathbf{x}_{1:k}|c_{1:k}, \mathbf{y}_{1:k})$ and $p(\alpha_{1:p}|c_{1:k}, a_{1:p})$ can be deduced exactly by use of the traditional Kalman filter equations. Thus the only space to search over and perform recursion upon is that defined by $p(c_{1:k}|\mathbf{y}_{1:k}, a_{1:p})$. This space is discrete but too large to enumerate all possible paths. Thus we turn to the approximation approach offered by particle filters.

By assuming that the distribution of c_k is dependent only upon $c_{1:k-1}$, $\mathbf{y}_{1:k}$ and $a_{1:p}$, the importance function can be factorised into terms such as $\pi(c_k|\mathbf{y}_{1:k}, a_{1:p}, c_{1:k-1})$. This allows recursion of the Rao-Blackwellised posterior

$$\begin{aligned}&p(c_{1:k}|\mathbf{y}_{1:k}, a_{1:p})\\ &\quad\propto p(\mathbf{y}_k, a_p|\mathbf{y}_{1:k-1}, a_{1:p-1}, c_{1:k})\\ &\qquad\times p(c_k|c_{k-1})\,p(c_{1:k-1}|\mathbf{y}_{1:k-1}, a_{1:p-1}),\end{aligned} \tag{23}$$

where

$$\begin{aligned}&p(\mathbf{y}_k, a_p|\mathbf{y}_{1:k-1}, a_{1:p-1}, c_{1:k})\\ &\quad= p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, c_{1:k})\\ &\qquad\times p(a_p|a_{1:p-1}, c_{1:k})\end{aligned} \tag{24}$$

and recursive updates to the weight are given by

$$w_k^{(i)} = w_{k-1}^{(i)} \times \frac{p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, c_{1:k}^{(i)})\,p(a_p|a_{1:p-1}, c_{1:k}^{(i)})\,p(c_k^{(i)}|c_{k-1}^{(i)})}{\pi(c_k^{(i)}|\mathbf{y}_{1:k}, a_{1:p}, c_{1:k-1}^{(i)})}. \tag{25}$$

For $k = 1$
  for $i = 1 : N$; draw $\mathbf{x}_1^{(i)}$, $\alpha_1^{(i)}$ and $c_1^{(i)}$ from respective priors
for $k = 2 :$ end
  for $i = 1 : N$
    Propagate particle i to a set, $s = \{1, \ldots, S\}$ of new locations $c_k^{(s)}$.
    Evaluate the new weight $w_k^{(s,i)}$ for each of these by propagating through the respective Kalman filter. This generates $\pi(c_k|\mathbf{y}_{1:k}, a_{1:p}, c_{1:k-1}^{(i)})$.
  for $i = 1 : N$
    Pick a new state for each particle from $\pi(c_k|\mathbf{y}_{1:k}, a_{1:p}, c_{1:k-1}^{(i)})$.
    Update weights according to (25).

Algorithm 1: Rao-Blackwellised particle filter.

The terms $p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, c_{1:k})$ and $p(a_p|a_{1:p-1}, c_{1:k})$ are calculated from the innovation vector and covariance of the respective Kalman filters (see (19) and (20)). $p(c_k|c_{k-1})$ is simplified to $p(c_k)$ and is hence the prior on score location as given in Section 5.

5.3. Algorithm

The algorithm therefore proceeds as given in Algorithm 1. At each iteration, each particle is propagated to a set S of new score locations and the probability of each is evaluated. Given the $N \times S$ set of potential states there are then two ways of choosing a new set of updated particles: either stochastic or deterministic selection. The first proceeds in a similar manner to that described by Cemgil [16] where for each particle the new state is picked from the importance function with a given probability. Deterministic selection simply takes the best N particles from the whole set of propagated particles. Fully stochastic resampling selection of the particles is not an optimal procedure in this case, as duplication of particles is wasteful. This leaves a choice between Cemgil's method of stochastically selecting one of the update proposals for each particle or the deterministic N-best approach. The latter has been adopted as intuitively sensible.

Particle filters suffer from degeneracy in that the posterior will eventually be represented by a single particle with high weight while many particles have negligible probability mass. Traditional PFs overcome this with resampling (see [23]) but both methods for particle selection in the previous section implicitly include resampling. However, degeneracy still exists, in that the PF will tend to converge to a single c_k state, so a number of methods were explored for increasing the diversity of the particles. Firstly, jitter [24] was added to the tempo process to increase local diversity. Secondly, a Metropolis-Hastings (MH) step [34] was used to explore jumps to alternative phases of the signal (i.e., to jump from tracking off-beat quavers to being on the beat). Also, an MH step to propose related tempos (i.e., doubling or halving the tracked tempo) was investigated but found to be counterproductive.

6. BEAT MODEL 2

The model described above formulates beat location as the free variable and time as a dependent, non-continuous variable, which seems counter-intuitive. Noting that the model is bilinear, a reformulation of the tempo process is thus presented now where time is the independent variable and tempo is modelled as Brownian motion[4] [35]. The state vector is now given by $\mathbf{z}_k = [\tau_k, \dot{\tau}_k]^T$ where τ_k is in beats and $\dot{\tau}_k$ is in beats per second (obviously related to bpm). Brownian motion, which is a limiting form of the random walk, is related to the tempo process by

$$d\dot{\tau}(t) = \sqrt{q}\, d\mathcal{B}(t) + \dot{\tau}(0), \tag{26}$$

where q controls the variance of the Brownian motion process $\mathcal{B}(t)$ (which is loosely the integral of a Gaussian noise process [32]) and hence the state evolution. This leads to

$$\tau(t) = \tau(0) + \int_0^t \dot{\tau}(s)ds. \tag{27}$$

Time t is now a continuous variable and hence $\tau(t)$ is also a continuously varying parameter, though only being "read" at algorithmic iterations k thus giving $\tau_k \triangleq \tau(t_k)$.

The new state equations are given by

$$\mathbf{z}_k = \Xi(\delta_k)\mathbf{z}_{k-1} + \beta_k, \tag{28}$$

$$\mathbf{y}_k = \Gamma_k t_k + \kappa_k, \tag{29}$$

where

$$t_k = t_0 + \sum_{j=1}^{k} \delta_k. \tag{30}$$

t_k is therefore the absolute time of an observation and δ_k is the inter-observation time. $\Xi(\delta_k)$ is the state update matrix and is given by

$$\Xi(\delta_k) = \begin{bmatrix} 1 & \delta_k \\ 0 & 1 \end{bmatrix}. \tag{31}$$

Γ_k acts in a similar manner to H_k in model one and is of variable length but is a vector of ones of the same length as $\mathbf{y}_k$. κ_k is modelled as zero mean Gaussian with covariance R_k as described above. β_k is modelled as zero mean Gaussian noise with covariance given as before by Bar-Shalom [32],

$$Q_k = q \begin{bmatrix} \frac{\delta_k^3}{3} & \frac{\delta_k^2}{2} \\ \frac{\delta_k^2}{2} & \delta_k \end{bmatrix}. \tag{32}$$

One of the problems associated with Brownian motion is that there is no simple, closed form solution for the prediction density, $p(t_k|\cdot)$. Thus attention is turned to

[4] Also termed as Wiener or Wiener-Levy process.

Initialise: $i = 1$; $\mathbf{z}_1 = \mathbf{z}_k$; X_k is the predicted inter-onset number of beats.
While $dt >$ tol,
  $i = i + 1$
  If $\max(\tau_{1:i}) < X_k$
    $dt = (\tau_{i-1} - X_k)/\dot{\tau}_{i-1}$
    Draw $\mathbf{z}_i \sim \mathcal{N}(\Xi_i \mathbf{z}_{i-1}, Q_i)$
    $t_i = t_{i-1} + dt$
  Else interpolate back
    Find I s.t. $\tau_I < X_k$ and $\tau_{I+1} > X_k$
    $t_e = t_I + (t_{I+1} - t_I) \times (X_k - \tau_I)/(\tau_{I+1} + \tau_I)$
    insert state J between I and $I + 1$
      $t_J = t_e$
      $dt = \min(t_{I+1} - t_e, t_e - t_I)$
      Draw $\mathbf{z}_J \sim \mathcal{N}(m, Q')$ where m and Q' are given below
Index $q = \min|(\tau_{1:i} - X_k)|$.
Return $\tau_k = X_k$, $t_k = t_q$ and $\dot{\tau}_k = \dot{\tau}_q$.

Algorithm 2: Sample hitting time.

an alternative method for drawing a *hitting time* sample of $\{t_k|\mathbf{z}_{k-1}, \tau_k = B, t_{k-1}\}$. This is an iterative process and, conditional upon initial conditions, a linear prediction for the time of the new beat is made. The system is then stochastically propagated for this length of time and a new tempo and beat position found. The beat position might under or overshoot the intended location. If it undershoots, the above process is repeated. If it overshoots, then an interpolation estimate is made conditional upon both the previous and subsequent data estimates. The iteration terminates when the error on τ_t falls below a threshold. At this point, the algorithm returns the hitting time t_k and the new tempo $\dot{\tau}_k$ at that hitting time. This is laid out explicitly in Algorithm 2, where Ξ_i is given by

$$\Xi_i = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \tag{33}$$

and Q_i by

$$Q_i = q \begin{bmatrix} \frac{dt^3}{3} & \frac{dt^2}{2} \\ \frac{dt^2}{2} & dt \end{bmatrix}. \tag{34}$$

$\mathcal{N}$ denotes the Gaussian distribution. The interpolation mean and covariance are given by [36]

$$\begin{aligned} Q' &= \left(Q_{I:J}^{-1} + \Xi_{J:I+1} Q_{J:I+1}^{-1} \Xi_{J:I+1}\right)^{-1}, \\ m &= Q'\left(Q_{I:J}^{-1} \Xi_{I:J} \mathbf{z}_I + \Xi_{J:I+1}^T Q_{J:I+1}^{-1} \mathbf{z}_{I+1}\right), \end{aligned} \tag{35}$$

where the index denotes the use of Ξ and Q from (33) and (34) with appropriate values of dt.

Thus, we now have a method of drawing a time t_k and new tempo $\dot{\tau}_k$ given a previous state $\mathbf{z}_{k-1}$ and proposed new score (beat) location τ_k. The algorithm then proceeds as before with a particle filter. The posterior can be updated, thus

$$\begin{aligned} p(\mathbf{z}_{1:k}, t_{1:k}|\mathbf{y}_{1:k}) \propto\ & p(\mathbf{y}_k|\mathbf{z}_{1:k}, t_{1:k}) p(t_k|t_{1:k-1}, \mathbf{z}_{1:k}) p(\mathbf{z}_k|\mathbf{z}_{1:k-1}) \\ & \times p(\mathbf{z}_{1:k-1}, t_{1:k-1}|\mathbf{y}_{1:k-1}), \end{aligned} \tag{36}$$

where $p(\mathbf{z}_k|\mathbf{z}_{1:k-1})$ can be factorised:

$$p(\mathbf{z}_k|\mathbf{z}_{1:k-1}) = p(\tau_k|\mathbf{z}_{k-1}) p(\dot{\tau}_k|\mathbf{z}_{k-1}, \tau_k). \tag{37}$$

Prior importance sampling [23] is used via the hitting time algorithm above to draw samples of $\dot{\tau}_k$ and t_k:

$$\pi(\mathbf{z}_k, t_k|\mathbf{z}_{1:k-1}, t_{1:k-1}, \mathbf{y}_{1:k}) = p(\dot{\tau}_k|\mathbf{z}_{k-1}, \tau_k) p(t_k|t_{1:k-1}, \mathbf{z}_{1:k}). \tag{38}$$

This leads to the weight update being given by

$$w_k^{(i)} = w_{k-1}^{(i)} \times p(\mathbf{y}_k|\mathbf{z}_{1:k}^{(i)}, t_{1:k}^{(i)}) p(\tau_k^{(i)}|\mathbf{z}_{k-1}^{(i)}). \tag{39}$$

As before in Section 5, a single beat is split into 24 subdivisions and a prior set upon these as given above in (17); $p(\tau_k|\mathbf{z}_{k-1})$ again reduces to $p(\tau_k) \equiv p(c_k)$. $p(\mathbf{y}_k|\mathbf{z}_{1:k}^{(i)}, t_{1:k}^{(i)})$ is the likelihood; if κ_k from (29) is modelled in the same way as ν_k from (14) then the likelihood is Gaussian with covariance again given by R_k which is diagonal and of the same dimension, j as the observation vector $\mathbf{y}_k$. Γ_k is then a $j \times 1$ matrix with all entries being 1.

Also as before, to explore the beat quantisation space $\tau_{1:k}$ effectively, each particle is predicted onward to S new positions for τ_k and therefore again, a set of $N \times S$ potential particles is generated. Deterministic selection in this setting is not appropriate so resampling is used to stochastically select N particles from the $N \times S$ set. This acts instead of the traditional resampling step in selecting high probability particles.

Amplitude modelling is also included in an identical form to that described in Section 5.1 which modifies (39) to

$$w_k^{(i)} = w_{k-1}^{(i)} \times p(\mathbf{y}_k|\mathbf{z}_{1:k}^{(i)}, t_{1:k}^{(i)}) p(a_p|\mathbf{z}_{1:k}^{(i)}, t_{1:k}^{(i)}) p(\tau_k^{(i)}|\mathbf{z}_{k-1}^{(i)}). \tag{40}$$

Also, the MH step described in Section 5.3 to explore different phases of the beat is used again.

7. RESULTS

The algorithms described above in Sections 5 and 6 have been tested on a large database of musical examples drawn from a variety of genres and styles, including rock/pop, dance, classical, folk and jazz. 200 samples, averaging about one minute in length were used and a "ground truth" manually generated for each by recording a trained musician clapping in time to the music.

The aim is to estimate the tempo and quantisation parameters over the whole dataset; in both models, the sequence of filtered estimates is not the best representation of this, due to locally unlikely data. Therefore, because each

Table 1: Results for beat tracking algorithms expressed as a total percentage averaged over the whole database.

	Raw		Allowed	
	C-L	TOT	C-L	TOT
Model 1	51.5	58.0	69.2	82.2
Model 2	34.1	38.4	54.4	72.8
Scheirer	26.8	41.9	33.0	53.4

particle maintains its own state history, the maximum a posteriori particle at the final iteration was chosen. The parameter sets used within each algorithm were chosen heuristically; it was deemed impractical to optimise them over the whole database. Various numbers of particles N were tried though results are given below for $N = 200$ and 500 for models one and two, respectively. Above these values, performance continued to increase very slightly, as one would expect, but computational effort also increased proportionally.

Tracking was deemed to be accurate if the tempo was correct (interbeat interval matches to within 10%) and a beat was located within 15% of the annotated beat location.[5] Klapuri [14] defines a measure of success as the longest consecutive region of beats tracked correctly as a proportion of the total (denoted "C-L" for consecutive-length). Also presented is a total percentage of correctly tracked beats (labelled "TOT"). The results are presented in Table 1. It was noted that the algorithms sometimes tracked at double or half tempo in psychologically plausible patterns; also, dance music with heavy off-beat accents often caused the algorithm to track 180° out of phase. The "allowed" columns of the table show results accepting these errors. Also shown for comparison are the results obtained using Scheirer's algorithm [7].

The current state of the art is the algorithm of Klapuri [14] with 69% success for longest consecutive sequence and 78% for total correct percentage (accepting errors) on his test database consisting of over 400 examples. Thus the performance of our algorithm is at least comparable with this.

Figure 2 shows the results for model one over the whole database graphically while Figure 3 shows the same for model two. These are ordered by style and then performance within the style category. Figure 4 shows the tempo profile for a correctly tracked example using model one; note the close agreement between the hand labelled data and the tracked tempo.

8. DISCUSSION

The algorithms described above have some similar elements but their fundamental operation is quite different: the Rao-Blackwellised model of Section 5 actually bears a significant resemblance to an interacting multiple models system of the type used in radar tracking [33], as many of the stages are actually deterministic. The second model, however, is much more typically a particle filter with mainly stochastic processes. Both have many underlying similarities in the model though the inference processes are significantly different.

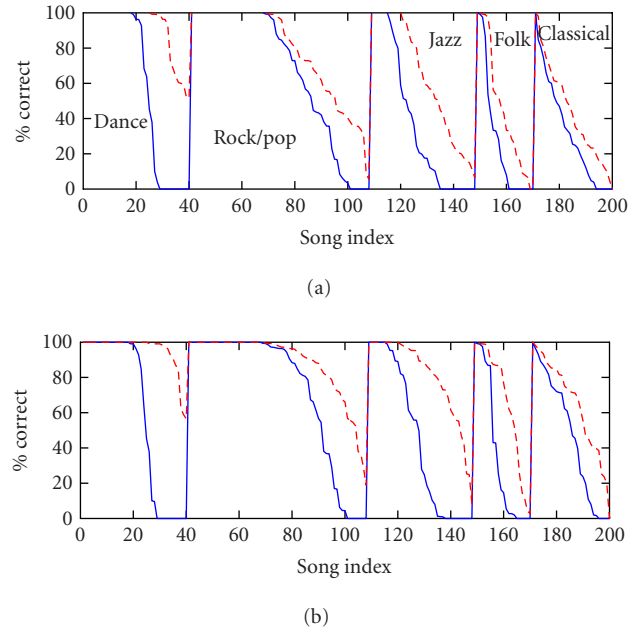

Figure 2: Results on test database for model one. The solid line represents raw performance and the dashed line is performance after acceptable tracking errors have been taken into account. (a) Maximum length correct (% of total). (b) Total percentage correct.

Thus, the results highlight some interesting comparisons between these two philosophies. On close examination, model two was better at finding the most likely local path through the data, though this was not necessarily the correct one in the long term. A fundamental weakness of the models is the prior on c_k (or equivalently, τ_k in model two) which intrinsically prefers higher tempos—doubling a given tempo places more onsets in metrically stronger positions which is deemed more likely by the prior given in (17). Because the stochastic resampling step efficiently selects and boosts high probability regions of the posterior, model two would often pick high tempos to track (150-200bpm) which accounts for the very low "raw" results.

A second problem also occurs in model two: because duplication of paths through the $\tau_{1:k}$ space is necessary to fully populate each quantisation hypothesis, fewer distinct paths are kept at each iteration. By comparison, the N-best selection scheme of model one ensures that each particle represents a unique $c_{1:k}$ set and more paths through the state space are kept for a longer lag. This allows model one to recover better from a region of poor data. This also provides an explanation for why model one does not track at high tempo so often—because more paths though the state-space are retained for longer, more time is allowed for the amplitude process to influence the choice of tempo mode. Thus, the conclusion is drawn that the first model is more attractive: the Rao-Blackwellisation of the tempo process allows the search of the quantisation space to be much more effective.

[5] The clapped signals were often slightly in error themselves.

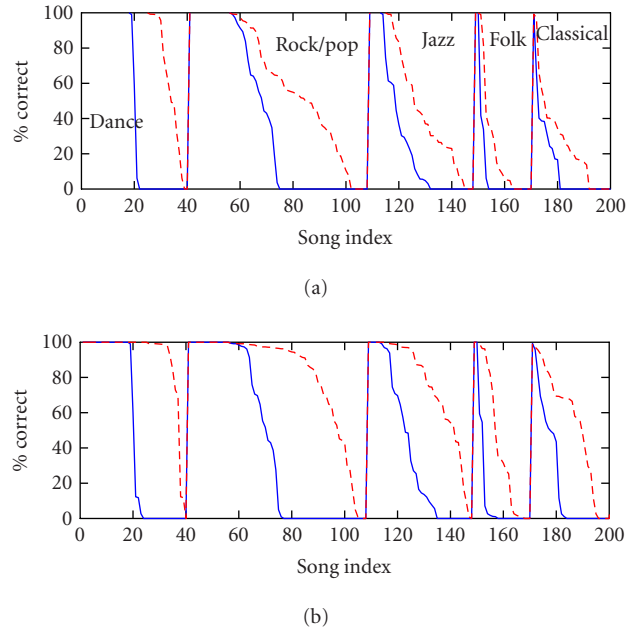

FIGURE 3: Results for model two. (a) Maximum length correct (% of total). (b) Total percentage correct.

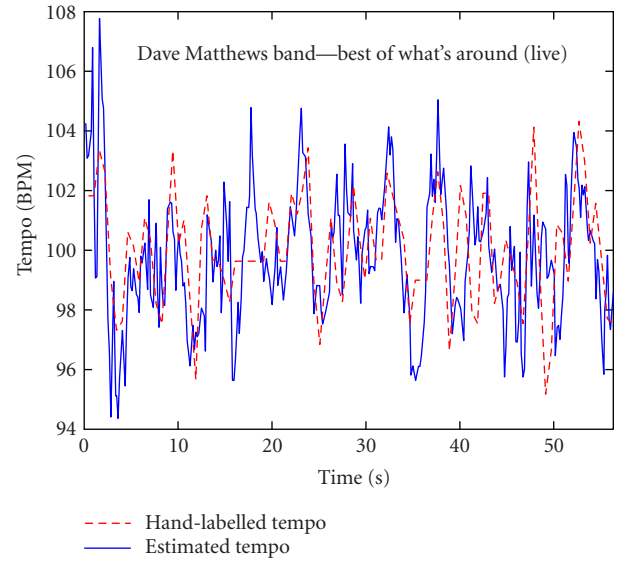

FIGURE 4: Tempo evolution for a correctly tracked example using model one.

The remaining lack of performance can be accredited to four causes: the first is tracking at multiple tempo modes—sometimes tracking fails at one mode and settles a few beats later into a second mode. The results only reflect one of these modes. Secondly, stable tracking sometimes occurs at psychologically implausible modes (e.g., 1.5 times the correct tempo) which are not included in the results above. The third cause is poor onset detection. Finally, there are also examples in the database which exhibit extreme tempo variation which is never followed.

The result of this is a number of suggestions for improvements: firstly, the onset detection is crucial and if the detected onsets are unreliable (especially at the start of an example) it is unlikely that the algorithm will ever be able to track the beat properly. This may suggest an "online" onset detection scheme where the particles propose onsets in the data, rather than the current offline, hard decision system. The other potential scheme for overcoming this would be to propose a salience measure (e.g., [21]) and directly incorporate this into the state evolution process, thus hoping to differentiate between likely and unlikely beat locations in the data; currently, the Rao-Blackwellised amplitude process has been given weak variances and hence has little effect in the algorithm, other than to propose correct phase. The other problems commonly encountered were tempo errors by plausible ratios; Metropolis-Hastings steps [27] to explore other modes of the tempo posterior were tried but have met with little success.

Thus it seems likely that any real further improvement will have to come from music theory incorporated into the algorithm directly, and in a style-specific way—it is unlikely that a beat tracker designed for dance music will work well on choral music! Thus, data expectations and also anticipated tempo evolutions and onset locations would have to be worked into the priors in order to select the correct tempo. This will probably result in an algorithm with many ad-hoc features but, given that musicians have spent the better part of 600 years trying to create music which confounds expectation, it is unlikely that a simple, generic model to describe all music will ever be found.

9. CONCLUSIONS

Two algorithms using particle filters for generic beat tracking across a variety of musical styles are presented. One is based upon the Kalman filter and is close to a multiple hypothesis tracker. This performs better than a more stochastic implementation which models tempo as a Brownian motion process. Results with the first model are comparable with the current state of the art [14]. However, the advantage of particle filtering as a framework is that the model and the implementation are separated allowing the easy addition of extra measures to discriminate the correct beat. It is conjectured that further improvement is likely to require music specific knowledge.

ACKNOWLEDGMENTS

This work was partly supported by the research program BLISS (IST-1999-14190) from the European Commission. The first author is grateful to the Japan Society for the Promotion of Science and the Grant-in-Aid for Scientific Research in Japan for their funding. The authors thank P. Comon and C. Jutten for helpful comments and are grateful to the anonymous reviewers for their helpful suggestions which have greatly improved the presentation of this paper.

REFERENCES

[1] C. Raphael, "A probabilistic expert system for automatic musical accompaniment," *J. Comput. Graph. Statist.*, vol. 10, no. 3, pp. 486–512, 2001.

[2] F. Gouyon, L. Fabig, and J. Bonada, "Rhythmic expressiveness transformations of audio recordings: swing modifications," in *Proc. Int. Conference on Digital Audio Effects Workshop*, London, UK, September 2003.

[3] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech, and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[4] E. D. Scheirer, "About this business of metadata," in *Proc. International Symposium on Music Information Retrieval*, pp. 252–254, Paris, France, October 2002.

[5] J. Seppänen, "Tatum grid analysis of musical signals," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 131–134, New Paltz, NY, USA, October 2001.

[6] S. W. Hainsworth, *Techniques for the automated analysis of musical audio*, Ph.D. thesis, Cambridge University Engineering Department, Cambridge, UK, 2004.

[7] E. D. Scheirer, "Tempo and beat analysis of acoustical musical signals," *J. Acoust. Soc. Amer.*, vol. 103, no. 1, pp. 588–601, 1998.

[8] E. W. Large and M. R. Jones, "The dynamics of attending: How we track time varying events," *Psychological Review*, vol. 106, no. 1, pp. 119–159, 1999.

[9] J. Foote and S. Uchihashi, "The beat spectrum: a new approach to rhythm analysis," in *Proc. IEEE International Conference on Multimedia and Expo*, pp. 881–884, Tokyo, Japan, August 2001.

[10] M. Goto, "An audio-based real-time beat tracking system for music with or without drum-sounds," *J. of New Music Research*, vol. 30, no. 2, pp. 159–171, 2001.

[11] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *J. of New Music Research*, vol. 30, no. 1, pp. 39–58, 2001.

[12] J. Laroche, "Estimating tempo, swing and beat locations in audio recordings," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 135–138, New Paltz, NY, USA, October 2001.

[13] C. Raphael, "Automated rhythm transcription," in *Proc. International Symposium on Music Information Retrieval*, pp. 99–107, Bloomington, Ind, USA, October 2001.

[14] A. Klapuri, "Musical meter estimation and music transcription," in *Proc. Cambridge Music Processing Colloquium*, pp. 40–45, Cambridge University, UK, March 2003.

[15] R. D. Morris and W. A. Sethares, "Beat tracking," in *7th Valencia International Meeting on Bayesian Statistics*, Tenerife, Spain, June 2002, personal communication with R. Morris.

[16] A. T. Cemgil and B. Kappen, "Monte Carlo methods for tempo tracking and rhythm quantization," *J. Artificial Intelligence Research*, vol. 18, no. 1, pp. 45–81, 2003.

[17] J. A. Bilmes, "Timing is of the essence: perceptual and computational techniques for representing, learning and reproducing expressive timing in percussive rhythm," M.S. thesis, Media Lab, MIT, Cambridge, Mass, USA, 1993.

[18] C. Drake, A. Penel, and E. Bigand, "Tapping in time with mechanical and expressively performed music," *Music Perception*, vol. 18, no. 1, pp. 1–23, 2000.

[19] D.-J. Povel and P. Essens, "Perception of musical patterns," *Music Perception*, vol. 2, no. 4, pp. 411–440, 1985.

[20] H. C. Longuet-Higgins and C. S. Lee, "The perception of musical rhythms," *Perception*, vol. 11, no. 2, pp. 115–128, 1982.

[21] R. Parncutt, "A perceptual model of pulse salience and metrical accent in musical rhythms," *Music Perception*, vol. 11, no. 4, pp. 409–464, 1994.

[22] M. J. Steedman, "The perception of musical rhythm and metre," *Perception*, vol. 6, no. 5, pp. 555–569, 1977.

[23] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[24] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings Part F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.

[25] A. F. M. Smith and A. E. Gelfand, "Bayesian statistics without tears: a sampling-resampling perspective," *Amer. Statist.*, vol. 46, no. 2, pp. 84–88, 1992.

[26] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[27] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Processing*, vol. 49, no. 3, pp. 613–624, 2001.

[28] G. Casella and C. P. Robert, "Rao-Blackwellisation of sampling schemes," *Biometrika*, vol. 83, no. 1, pp. 81–94, 1996.

[29] C. Duxbury, M. Sandler, and M. Davies, "A hybrid approach to musical note detection," in *Proc. 5th Int. Conference on Digital Audio Effects Workshop*, pp. 33–38, Hamburg, Germany, September 2002.

[30] S. Abdallah and M. Plumbley, "Unsupervised onset detection: a probabilistic approach using ICA and a hidden Markov classifier," in *Proc. Cambridge Music Processing Colloquium*, Cambridge, UK, March 2003.

[31] S. W. Hainsworth and M. D. Macleod, "Onset detection in musical audio signals," in *Proc. International Computer Music Conference*, pp. 163–166, Singapore, September–October 2003.

[32] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, vol. 179 of *Mathematics in Science and Engineering*, Academic Press, Boston, Mass, USA, 1988.

[33] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, Norwood, Mass, USA, 1999.

[34] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, Eds., *Markov chain Monte Carlo in practice*, Chapman & Hall, London, UK, 1996.

[35] B. Øksendal, *Stochastic Differential Equations*, Springer-Verlag, New York, NY, USA, 3rd edition, 1992.

[36] M. Orton and A. Marrs, "Incorporation of out-of-sequence measurements in non-linear dynamic systems using particle filters," Tech. Rep., Cambridge University Engineering Department, Cambridge, UK, 2001.

Stephen W. Hainsworth was born in 1978 in Stratford-upon-Avon, England. During 8 years at the University of Cambridge, he was awarded the B.A. and M.Eng. degrees in 2000 and the Ph.D. in 2004, with the latter concentrating on techniques for the automated analysis of musical audio. Since graduating for the third time, he has been working in London for Tillinghast-Towers Perrin, an actuarial consultancy.

Malcolm D. Macleod was born in 1953 in Cathcart, Glasgow, Scotland. He received the B.A. degree in 1974, and Ph.D. on discrete optimisation of DSP systems in 1979, from the University of Cambridge. From 1978 to 1988 he worked for Cambridge Consultants Ltd, on a wide range of signal processing, electronics, and software research and development projects. From 1988 to 1995 he was a Lecturer in the Signal Processing and Communications Group, the Engineering Department of Cambridge University, and from 1995 to 2002 he was the Department's Director of Research. In November 2002 he joined the Advanced Signal Processing Group at QinetiQ, Malvern, as a Senior Research Scientist. He has published many papers in the fields of digital filter design, nonlinear filtering, adaptive filtering, efficient implementation of DSP systems, optimal detection, high-resolution spectrum estimation and beamforming, image processing, and applications in sonar, instrumentation, and communication systems.